MDPC-McEliece: New McEliece Variants from Moderate Density Parity-Check Codes

Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, Paulo S. L. M. Barreto

MDPC-McEliece: New McEliece Variants from Moderate Density Parity-Check Codes

Rafael Misoczki
Project SECRET
INRIA-Rocquencourt, France
Rafael.Misoczki@inria.fr

Jean-Pierre Tillich
Project SECRET
INRIA-Rocquencourt, France
Jean-Pierre.Tillich@inria.fr

Nicolas Sendrier
Project SECRET
INRIA-Rocquencourt, France
Nicolas.Sendrier@inria.fr

Paulo S. L. M. Barreto
Escola Politécnica
Univ. de São Paulo, Brazil
PBarreto@usp.br

***Abstract*—In this work, we propose two McEliece variants: one from Moderate Density Parity-Check (MDPC) codes and another from quasi-cyclic MDPC codes. MDPC codes are LDPC codes of higher density (and worse error-correction capability) than what is usually adopted for telecommunication applications. However, in cryptography we are not necessarily interested in correcting many errors, but only a number which ensures an adequate security level. By this approach, we reduce under certain hypotheses the security of the scheme to the well studied decoding problem. Furthermore, the quasi-cyclic variant provides extremely compact-keys (for 80-bits of security, public-keys have only 4801 bits).**

I. Introduction

Cryptosystems based on the hardness of factoring or discrete logarithm can be quantum attacked in polynomial time [1]. This threatens most if not all public-key cryptosystems deployed in practice, such as RSA. Code-based cryptography is believed to be quantum resistant and is therefore considered as a viable replacement. Furthermore, it provides better algorithmic complexity than traditional schemes.

The first code-based cryptosystem is the McEliece cryptosystem [2], originally proposed using Goppa codes. Its security is based on two assumptions, the indistinguishability of the code family and the hardness of decoding a generic linear code. The decoding problem is a well studied NP-complete problem [3], believed to be hard after decades of research. On the other hand, the indistinguishability problem is usually the weakest one, strongly depending on the choice of the code family. For example, a distinguisher for high rate Goppa codes has been presented in [4]. Although this does not represent a practical attack, it might suggest that Goppa codes are not the optimal choice for code-based cryptography.

The main drawback of this scheme is its large keys. Recently, proposals using codes with a large automorphism group (e.g. quasi-cyclic [5], [6] or quasi-dyadic codes [7]) allowed to reduce the key size. However the redundancy added to these algebraic codes allowed to break [8] many of them (except the binary case of [7]). This kind of attack is exponential in nature and can be prevented by choosing more conservative parameters. However note that codes which have no algebraic structure would prevent this threat.

Related work. Low-Density Parity Check (LDPC) codes [9] are good candidates for this scenario. These are codes with no algebraic structure which meet a very simple combinatorial property: they admit a sparse parity-check matrix. They have been repeatedly suggested for the McEliece scheme [10], [11], [12]. The main problem of using LDPC codes in this context is that their low weight parity-check rows can be seen as low weight codewords in the dual of the public code. Thus a straightforward attack is to search for dual low-weight codewords to build a sparse parity-check able to decode efficiently. As shown in [10], the extremely low row weights suggested for LDPC codes allow the effectiveness of this attack. In [11], to avoid this attack, the authors increased the weight of the dual codewords replacing the permutation matrix used in the scheme by a sparse invertible matrix of some small constant row weight. Nevertheless, the unfortunate choices for this new matrix (among other properties) allowed to successfully cryptanalyze the scheme [13]. In [12], a more general construction has been able to thwart this attack. Furthermore, the authors suggested a quasi-cyclic structure reducing the public-keys to 48384 bits[1].

Our contribution. Our first observation is that any auxiliary matrix of constant row weight is needed to instantiate the McEliece scheme with LDPC codes. Simply increasing moderately the length and the row weight of the secret sparse parity-check matrix is enough to avoid all known message and key recovery attacks. We call these codes Moderate Parity Check (MDPC) codes[2] and use them to instantiate the

Paulo S. L. M. Barreto is supported by the Brazilian National Council for Scientific and Technological Development (CNPq) under universal grant 482342/2011-0 and research productivity grant 306935/2012-0.

[1]The authors did not consider CCA-2 secure conversions, which would allow public-keys in systematic form, reducing the key size to 12096 bits.

[2]In [14], the terminology MDPC is used for the same concept. It is shown that certain QC-MDPC codes may perform well at moderate lengths by using a variation of the standard belief propagation concept.

McEliece scheme. We give a quite satisfactory security reduction towards the well studied syndrome decoding problem. To achieve this goal, we make a single, natural assumption: distinguishing an MDPC code from a random linear code amounts to being able to ascertain the existence of low weight codewords in its dual code. This provides a strong security argument. Besides, a quasi-cyclic structure provides extremely compact keys. The extended-version of this work is [15].

II. Preliminaries

Definition 1 (Linear codes): The (Hamming) weight of a vector $x \in \mathbb{F}_2^n$ is the number $\mathsf{wt}(x)$ of its nonzero components. A *binary* (n, r)*-linear code* $\mathcal{C}$ of length n, dimension $n - r$ and codimension r, is a $(n - r)$-dimensional vector subspace of $\mathbb{F}_2^n$. It is spanned by the rows of a matrix $G \in \mathbb{F}_2^{(n-r)\times n}$, called a *generator matrix* of $\mathcal{C}$. Equivalently, it is the kernel of a matrix $H \in \mathbb{F}_2^{r\times n}$, called a *parity-check matrix* of $\mathcal{C}$. The *codeword* $c \in \mathcal{C}$ of a vector $m \in \mathbb{F}_2^{(n-r)}$ is $c = mG$. The *syndrome* $s \in \mathbb{F}_2^r$ of a vector $e \in \mathbb{F}_2^n$ is $s = He^T$. The *dual* $\mathcal{C}^\perp$ of $\mathcal{C}$ is the linear code spanned by the rows of any parity-check matrix of $\mathcal{C}$.

Definition 2 (Quasi-cyclic code): An (n, r)-linear code is quasi-cyclic (QC) if there is some integer n_0 such that every cyclic shift of a codeword by n_0 places is again a codeword.

When $n = n_0 p$, for some integer p, it is possible and convenient to have both generator and parity check matrices composed by $p \times p$ circulant blocks. A circulant block is completely described by its first row (or column) and the algebra of $p \times p$ binary circulant matrices is isomorphic to the algebra of polynomials modulo $x^p - 1$ over $\mathbb{F}_2$, enabling efficient computation.

Definition 3 (LDPC/MDPC codes): An (n, r, w)-LDPC or MDPC code is a linear code of length n, codimension r admitting a parity-check matrix with constant row weight w.

LDPC and MDPC codes only differ in the row weight w. While LDPC codes have small constant row weights (usually less than 10), we assume for MDPC codes row weights which scale in $O(\sqrt{n \log n})$. When they are also quasi-cyclic, we call them (n, r, w)-QC-LDPC or QC-MDPC codes.

III. Moderate Density Parity-Check McEliece

We present the construction of our codes and our variants.

A. (n, r, w)-MDPC code construction

1) Generate r vectors $(h_i \in \mathbb{F}_2^n)_{0\le i<r}$, of weight w at random.
2) The (n, r, w)-MDPC code is defined by a parity-check matrix $H \in \mathbb{F}_2^n$ of i-th row h_i.

B. (n, r, w)-QC-MDPC code construction

1) Generate a vector $h \in \mathbb{F}_2^n$ of weight w at random.
2) The (n, r, w)-QC-MDPC code is defined by a quasi-cyclic parity-check matrix $H \in \mathbb{F}_2^n$ of first row h.
3) The other $r - 1$ rows of H are obtained from the $r - 1$ quasi-cyclic shifts of h.

For the MDPC construction, with overwhelming probability H is of full rank and the rightmost $r \times r$ block is invertible after possibly swapping a few columns. For the QC-MDPC construction, let $n = n_0 p$, for n_0, $p \in \mathbb{Z}^*$. We construct a matrix $H = [H_0|H_1|\dots|H_{n_0-1}] \in \mathbb{F}_2^{r\times n}$ formed by one row of n_0 circulant blocks H_i of size $p \times p$. Each block H_i has row weight w_i, such that $w = \sum_{i=0}^{n_0-1} w_i$. Assuming H_{n_0-1} is non-singular, a generator matrix G in row reduced echelon form is easily derived from the H_i's blocks: on the left side, G has an identity block of size $(n - r) \times (n - r)$ and on the right side, a column of $n_0 - 1$ circulant blocks of size $p \times p$ obtained from: $(H_{n_0-1}^{-1} \cdot H_i)^T$, for $i \in [0..(n_0 - 2)]$.

C. (QC)-MDPC McEliece variant

1) Key-Generation.
   a) Generate a parity-check matrix $H \in \mathbb{F}_2^{r\times n}$ of a t-error-correcting (n, r, w)-MDPC or (n, r, w)-QC-MDPC code.
   b) Generate its corresponding generator matrix $G \in \mathbb{F}_2^{(n-r)\times n}$ in row reduced echelon form.
   c) The public key is G and the private key H.
2) Encryption. To encrypt $m \in \mathbb{F}_2^{(n-r)}$ into $x \in \mathbb{F}_2^n$:
   a) Generate $e \in \mathbb{F}_2^n$ of $\mathsf{wt}(e) \le t$ at random.
   b) Compute $x \leftarrow mG + e$.
3) Decryption. Let Ψ_H be a t-error correcting LDPC decoding algorithm equipped with the knowledge of the private H. To decrypt $x \in \mathbb{F}_2^n$ into $m \in \mathbb{F}_2^{(n-r)}$:
   a) Compute $mG \leftarrow \Psi_H(mG + e)$.
   b) Extract the plaintext m from the first $(n-r)$ positions of mG.

Note that this description gets rid[3] of the usual scrambling matrix S and permutation matrix P. Note also that the use of a CCA-2 security-conversion, e.g. [16], allows for G in systematic-form, without bringing any security-flaw. The public key-size of the QC-MDPC variant is $(n-r)$ and the MDPC is $r(n - r)$. Regarding the QC case, note that a quasi-cyclic structure, by itself, does not imply a significant improvement for attacks. All previous attacks on compact-keys McEliece variants are based on the *combination* of a quasi-cyclic/dyadic structure and some *algebraic* code structure.

IV. Decoding MDPC codes

Our MDPC codes will be decoded with a variant of the Gallager's bit flipping algorithm [9]. This iterative decoding algorithm provides an error-correction capability for LDPC codes which increases linearly with the code-length and

[3] A folklore reasoning assigns security functions to those matrices. However it is enough that the public-key does not reveal any useful information for decoding, a condition satisfied by the dense public matrix.

decreases more or less linearly with the weight w of the parity-checks. Thus, when moving from LDPC to MDPC codes, a degradation in the error-correcting capability is expected. However in cryptography we are not necessarily interested in correcting a large number of errors, but only a number which ensures an adequate security level.

Gallager's bit flipping algorithm works as follows. At each iteration, the number of unsatisfied parity-check equations associated to each bit of the message is computed. Each bit associated to more than b unsatisfied equations is flipped and the syndrome is recomputed. This process is repeated until either the syndrome becomes zero or after a maximum number of iteration. It is easy to see that this algorithm has complexity $O(nwI)$, where I stands for the average number of iterations. Due to the increased row weight (and the existence of short-cycles in the corresponding Tanner graph), MDPC codes may lead to an increased number of iterations. To minimize this problem, we suggest a variant of Gallager's algorithm changing the choice for b. Below a few possibilities for b:

I. Precomputing b (see inequality 4.16, pg. 46, [9]);
II. In [17], b is chosen as Max_{upc}, the maximum number of unsatisfied parity-check equations;
III. Our approach: $b = \text{Max}_{\text{upc}} - \delta$, for a small integer δ.

Approach II is more general than I, leading to a better error-correcting capability at the price of an increased number of iterations. Approach III combines the benefits of I and II since it reduces the overall number of iterations obtained by Approach II (much more bits are flipped at each iteration) and it provides an error-correcting capability as good as Approach II. This last benefit is due to the following strategy. Every time the algorithm fails to decode, the value of δ is decreased by 1 and the process is restarted. Obviously when $\delta = 0$, we are back to Approach II. Regarding the parameters suggested in Section VI, a good choice for δ is around 5, reducing the number of iterations from ~ 65 to less than 10.

To estimate the error correction capability of Gallager's algorithm for MDPC codes we use Gallager's analysis [9] which gives a threshold for the number of errors that an (n, r, w)-LDPC code may correct. Although this analysis is not quite precise for MDPC codes (due to the short cycles in the associated Tanner graph), it provides an upper bound on its error correction capability. Then through exhaustive simulation is possible to estimate the quality of an MDPC code in terms of its *decoding failure rate* (DFR). Thus it suffices to decrease the number of errors from this threshold until achieving an adequate DFR. For example, the parameters of Section VI reach a DFR of at most 10^{-7}.

Notice that, in cryptography, this non-zero probability of decoding failure must be treated. A straightforward approach is to conservatively choose the number of errors so that the DFR is negligible (e.g. smaller than the machine failure rate). A second and on-the-fly approach is to switch to more sophisticated algorithms, like [18], which achieves better error-correction capability at the price of a significantly increased decoding complexity. Finally, when the application allows, a third approach consists in using a CCA-2 secure conversion [16], ensuring the indistinguishability of the encrypted messages. Then, in the case of a decoding failure, new encryptions can be requested and the adversary would not be able to extract any information from this redundancy since the encrypted messages behave like random sequences.

V. Security Assessment

A. Security reduction

By security reduction, we mean a proof that an adversary able to attack the scheme is able solve some (presumably hard) algorithmic problem with a similar computational effort. We start giving the description of the generic reduction adapted from [19] for the Niederreiter cryptosystem [20]. It is easy to see that this security reduction also holds for the McEliece scheme (which is equivalent in terms of security to the Niederreiter scheme [21]) at the price of a more involved probability space and statements. Next, the generic reduction and then the discussion regarding our proposal.

Let $\mathcal{F}_{n,r,w}$ denote a t-error correcting code family which can be either (n, r, w)-MDPC or QC-MDPC (the statements are valid in both cases). The public key is a parity check matrix of some code in $\mathcal{F}_{n,r,w}$. The key space of $\mathcal{F}_{n,r,w}$ is $\mathcal{K}_{n,r,w}$ and the *apparent* key space of $\mathcal{F}_{n,r,w}$ is $\mathcal{H}_{n,r} \supset \mathcal{K}_{n,r,w}$. In the MDPC case, $\mathcal{H}_{n,r}$ is the set of full rank matrices in $\mathbb{F}_2^{r\times n}$. For QC-MDPC, $\mathcal{H}_{n,r}$ is restricted to block circulant matrices.

Generic Reduction. Let $\mathcal{S}_n(0, t)$ denote the sphere centered in zero of radius t in the Hamming space $\mathbb{F}_2^n$ and let Ω denote the probability space consisting of the sample space $\mathcal{H}_{n,r}\times\mathcal{S}_n(0, t)$ equipped with uniform distribution. We define:

- **Distinguisher**: A program $\mathcal{D} : \mathcal{H}_{n,r} \longrightarrow \{0, 1\}$ is a (T, ϵ)-*distinguisher* for $\mathcal{K}_{n,r,w}$ (*vs.* $\mathcal{H}_{n,r}$) if it runs in time at most T and the advantage of $\mathcal{D}$ for $\mathcal{K}_{n,r,w}$ given by $Adv(\mathcal{D}, \mathcal{K}_{n,r,w}) = |Pr_\Omega(\mathcal{D}(H) = 1|H \in \mathcal{K}_{n,r,w}) - Pr_\Omega(\mathcal{D}(H) = 1)|$ is greater than ϵ.
- **Decoder**: A program $\phi : \mathcal{H}_{n,r} \times \mathbb{F}_2^r \longrightarrow \mathcal{S}_n(0, t)$ is a (T, ϵ)-*decoder* for $(\mathcal{H}_{n,r}, t)$ if it runs in time at most T and its success probability given by $Succ(\phi) = Pr_\Omega(\phi(H, eH^\intercal) = e)$ is greater than ϵ.
- **Adversary**: A program $\mathcal{A} : \mathcal{H}_{n,r} \times \mathbb{F}_2^n \longrightarrow \mathcal{S}_n(0, t)$ is a (T, ϵ)-*adversary* against $\mathcal{K}_{n,r,w}$-Niederreiter if it runs in time at most T and its success probability given by $Succ(\mathcal{A}, \mathcal{K}_{n,r,w}) = Pr_\Omega(\mathcal{A}(H, eH^\intercal) = e|H \in \mathcal{K}_{n,r,w})$ is greater than ϵ.

A distinguisher for $\mathcal{K}_{n,r,w}$ *vs.* $\mathcal{H}_{n,r}$ and a decoder for $(\mathcal{H}_{n,r}, t)$ provide solutions respectively to the *Code distinguishing* and to the *Computational syndrome decoding problem*. Below we present Proposition 1 showing that if none of those problems can be solved efficiently then no efficient

adversary against the scheme exists.

- *Problem 1.* (**Code distinguishing problem**). Given the parameters $\mathcal{K}_{n,r,w}$, $\mathcal{H}_{n,r}$ and the instance $H \in \mathcal{H}_{n,r}$, is $H \in \mathcal{K}_{n,r,w}$?
- *Problem 2.* (**Computational syndrome decoding problem**). Given the parameters $\mathcal{H}_{n,r}$, an integer $t > 0$ and the instance $H \in \mathcal{H}_{n,r}$, $s \in \mathbb{F}_2^r$, find a vector $e \in \mathcal{S}_n(0,t)$ such that $eH^{\mathrm{T}} = s$.

Proposition 1: Given the parameters (n,r,w) and t, if there exists a (T,ϵ)-adversary against $\mathcal{K}_{n,r,w}$-Niederreiter, then there exists either a $(T,\epsilon/2)$-decoder for $(\mathcal{H}_{n,r},t)$ or a $(T+O(n^2),\epsilon/2)$-distinguisher for $\mathcal{K}_{n,r,w}$ *vs.* $\mathcal{H}_{n,r}$.

Proof: in [15]. ■

The MDPC and the QC-MDPC cases. We introduce an additional problem which consists in deciding the existence of words of given weight in a given linear code. Note that the code we consider below has a *generator matrix* $H \in \mathcal{H}_{n,r}$, it is thus the dual of a code in $\mathcal{F}_{n,r,w}$.

- *Problem 3.* (**Codeword existence problem**). Given the parameters $\mathcal{H}_{n,r}$, an integer $w > 0$ and the instance $H \in \mathcal{H}_{n,r}$, is there a codeword of weight w in the code of generator matrix H?

Ideally, we would like to replace Problem 1 by Problem 3 in Proposition 1. Unfortunately, one would need to replace the distinguisher advantage by the quantity: $Adv(\mathcal{E},\mathcal{K}_{n,r,w}) = |Pr_\Omega(\mathcal{E}(H)=1|H \in \mathcal{K}_{n,r,w}) - Pr_\Omega(\mathcal{E}(H)=1)|$, where $\mathcal{E}$ denotes a program deciding the existence of a word of weight w in a given code. However this quantity is not directly related to the hardness of Problem 3. Thus we reach our purpose if the following conjecture holds.

Conjecture 1: Solving Problem 1 for $(\mathcal{H}_{n,r},\mathcal{K}_{n,r,w})$ is not easier than solving Problem 3 for $(\mathcal{H}_{n,r},w)$.

Within this conjecture we could modify the reduction to a claim that the $\mathcal{K}_{n,r,w}$-McEliece scheme is at least as hard as either Problem 2 and Problem 3. Now if we remark that Problem 3 is polynomially equivalent to its associate computational problem (Problem 4), and that this Problem 4 is polynomially equivalent to Problem 2 (see [15]), we may then produce a strong security statement.

- *Problem 4.* (**Codeword finding problem**). Given the parameters $\mathcal{H}_{n,r}$, an integer $w > 0$, and the instance $H \in \mathcal{H}_{n,r}$, find a codeword of weight w in the code of generator matrix H.

Proposition 2: Assuming Conjecture 1, breaking the (QC)-MDPC variant of McEliece or Niederreiter is not easier than solving the syndrome decoding problem in a random (QC) linear code.

Proof: directly from Proposition 1 and the polynomial equivalence of problems 3–4 and 4–2 (see [15]). ■

B. Practical security

Consider the McEliece (or Niederreiter) scheme with an (n,r,w)-(QC)-MDPC code correcting t errors. We denote $\mathcal{C}$ the hidden (QC)-MDPC code defined by the public generator matrix of $\mathcal{C}$. We assume that the key distinguishing attack is equivalent to exhibit one codeword of $\mathcal{C}^\perp$ of weight w. The key recovery attack is equivalent to exhibit r codewords of $\mathcal{C}^\perp$ of weight w. The decoding attack is equivalent to decode t errors in an (n,r)-linear code. For all those attacks we have to solve either Problem 2 or Problem 4. For both, the best technique is information set decoding (ISD) [22]. In today's state-of-the-art the best variants derive from Stern's collision decoding algorithm [23]. There have been numerous improvements until the recent asymptotic result [24]. We have analyzed all of them and an unpublished non-assymptotic analyses of [24] gives slightly lower workfactors (see [15]).

We denote by $\mathrm{WF}_{\mathrm{isd}}(n,r,t)$ the cost for decoding t errors (or finding a codeword of weight t) in a (n,r)-binary linear code when there is a single solution of the problem. In short, ISD algorithms assume a pattern for the sought error vector and it analyzes a certain set of candidates until a solution is found. This set of candidates is stored in lists of a size $\mathcal{L}$ and each candidate has a probability P to produce the solution. When the parameters algorithm are *optimal*, the workfactor $\mathrm{WF}_{\mathrm{isd}}(n,r,t)$ is equal, up to a small factor, to the ratio L/P. The *Decoding One Out of Many* setting (DOOM) [25] analyzes the gains when the decoding problem have multiple solutions and the attacker is satisfied with a single solution. This work shows that when the problem has N_s solutions, the probability of success P increases by a factor N_s (as long as $N_sP \ll 1$) and when N_i instances are treated simultaneously the list size L increases at most by a factor $\sqrt{N_i}$. Thus the DOOM technique [25] provides a gain[4] of $N_s/\sqrt{N_i}$.

For our variants, the key distinguishing leads to $N_i = 1$, the zero syndrome, and $N_s = r$, the r parity-check rows. For key recovery, $N_s = r$ again and, in the QC-MDPC case, one word is enough. For QC-MDPC decoding, any cyclic shift of the syndrome is a valid instance (up to a block-wise cyclic shift), thus $N_i = N_s = r$. Table I summarizes these gains. *Example.* Let $n_0 = 2$, $n = 9602$, $r = 4801$, $w = 90$, $t = 84$, quasi-cyclic. The analysis of [24] gives costs of $2^{92.70}$ for key-recovery and $2^{87.16}$ for decoding. Decreasing them by factors of r and $\sqrt{r}$, the final costs are: $2^{80.47}$, $2^{81.04}$. A final remark on practical security: we choose r as a prime number to avoid attacks exploiting non-prime quasi-cyclicity [26], [27].

VI. Conclusion

MDPC codes seem to be very convenient for cryptographic purposes. They reduce the distinguishing problem to a well studied coding-theory problem: decoding linear codes. Besides, adding a quasi-cyclic structure, it provides extremely

[4] In general, the real gain is in fact slightly smaller because these algorithms depend on optimal parameters which are not the same for multiple instances.

	MDPC	QC-MDPC
Key distinguishing	$\frac{1}{r}\mathrm{WF}_{\mathrm{isd}}(n, n-r, w)$	$\frac{1}{r}\mathrm{WF}_{\mathrm{isd}}(n, n-r, w)$
Key recovery	$\mathrm{WF}_{\mathrm{isd}}(n, n-r, w)$	$\frac{1}{r}\mathrm{WF}_{\mathrm{isd}}(n, n-r, w)$
Decoding	$\mathrm{WF}_{\mathrm{isd}}(n, r, t)$	$\frac{1}{\sqrt{r}}\mathrm{WF}_{\mathrm{isd}}(n, r, t)$

TABLE I. BEST ATTACKS FOR CODE-BASED ENCRYPTION SCHEMES USING t-ERROR CORRECTING (n, r, w)-MDPC (OR QC-MDPC) CODES

TABLE II. QC-MDPC MCELICE SUGGESTED PARAMETERS. SYNDROME (r) AND KEY SIZE GIVEN IN BITS.

Level security	n_0	n	r	w	t	key-size
80	2	9602	4801	90	84	4801
80	3	10779	3593	153	53	7186
80	4	12316	3079	220	42	9237
128	2	19714	9857	142	134	9857
128	3	22299	7433	243	85	14866
128	4	27212	6803	340	68	20409
256	2	65542	32771	274	264	32771
256	3	67593	22531	465	167	45062
256	4	81932	20483	644	137	61449

compact keys. Table II presents parameters for the QC-MDPC variant and Table III a key-size comparison to [12] (keys in systematic form), [7] and [28]. Regarding its complexity, the encryption and key-generation are reduced to simple QC-block products, and the decryption takes less than 3ms in a non-optimized C++ implementation running at an Intel Xeon CPU @3.20GHz (80-bits sec. parameters). Note the system can be scaled to meet arbitrarily large security [15].

REFERENCES

[1] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM J. Comput.*, vol. 26, no. 5, pp. 1484–1509, 1997.

[2] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory," *Deep Space Network Progress*, vol. 44, pp. 114–116, 1978.

[3] E. Berlekamp, R. McEliece, and H. van Tilborg, "On the inherent intractability of certain coding problems (corresp.)," *Information Theory, IEEE Transactions on*, vol. 24, no. 3, pp. 384 – 386, may 1978.

[4] J.-C. Faugère, V. Gauthier, A. Otmani, L. Perret, and J.-P. Tillich, "A distinguisher for high rate McEliece cryptosystems," in *ITW 2011*, Paraty, Brazil, Oct. 2011, pp. 282–286.

[5] P. Gaborit, "Shorter keys for code based cryptography," in *International Workshop on Coding and Cryptography – WCC'2005*. Bergen, Norway: ACM Press, 2005, pp. 81–91.

[6] T. P. Berger, P.-L. Cayrel, P. Gaborit, and A. Otmani, "Reducing key length of the McEliece cryptosystem," in *Progress in Cryptology, Africacrypt'2009*, ser. LNCS, vol. 5580. Springer, 2009, pp. 77–97.

[7] R. Misoczki and P. S. L. M. Barreto, "Compact McEliece keys from Goppa codes," in *Selected Areas in Cryptography*, 2009, pp. 376–392.

TABLE III. KEY-SIZE COMPARISON. KEY-SIZES GIVEN IN BITS.

Level security	QC-MDPC	QC-LDPC [12]	QD-Goppa [7]	Goppa [28]
80	4801	12096	20480	460 647
128	9857	–	32768	1 537 536
256	32771	–	65536	7 667 855

[8] J.-C. Faugère, A. Otmani, L. Perret, and J.-P. Tillich, "Algebraic cryptanalysis of McEliece variants with compact keys," in *Advances in Cryptology, Eurocrypt'2010*, ser. LNCS, vol. 6110. Springer, 2010, pp. 279–298.

[9] R. G. Gallager, *Low-Density Parity-Check Codes*. M.I.T. Press, 1963.

[10] C. Monico, J. Rosenthal, and A. Shokrollahi, "Using low density parity check codes in the McEliece cryptosystem," in *IEEE International Symposium on Information Theory – ISIT'2000*. IEEE, 2000, p. 215.

[11] M. Baldi and F. Chiaraluce, "Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes," in *IEEE Int. Symposium on Information Theory ISIT'07*, 2007, pp. 2591 –2595.

[12] M. Baldi, M. Bodrato, and F. Chiaraluce, "A new analysis of the McEliece cryptosystem based on QC-LDPC codes," in *6th Int. Conf. on Sec. and Cryptography for Networks*. Springer, 2008, pp. 246–262.

[13] A. Otmani, J. Tillich, and L. Dallot, "Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes," *Special Issues of Mathematics in Computer Science*, vol. 3, no. 2, pp. 129–140, Jan. 2010.

[14] S. Ouzan and Y. Be'ery, "Moderate-density parity-check codes," *CoRR*, vol. abs/0911.3262, 2009.

[15] R. Misoczki, J.-P. Tillich, N. Sendrier, and P. S. L. M. Barreto, "MDPC-McEliece: New McEliece variants from moderate density parity-check codes," 2012. [Online]. Available: http://eprint.iacr.org/2012/409

[16] K. Kobara and H. Imai, "Semantically secure mceliece public-key cryptosystems -conversions for mceliece pkc -," in *Public Key Crypt.*, ser. LNCS, K. Kim, Ed. Springer, 2001, vol. 1992, pp. 19–35.

[17] W. Huffman and V. Pless, *Fundamentals of Error-Correcting Codes*. Cambridge University Press, 2003.

[18] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *Information Theory, IEEE Transactions on*, vol. 42, no. 2, pp. 429 – 445, march 1996.

[19] N. Sendrier, "On the use of structured codes in code based cryptography," in *Coding Theory and Cryptography III*, ser. Contactforum, S. Nikova, B. Preneel, and L. Storme, Eds. Koninklijke Vlaamse Acad. van België voor Wetenschaeppen en Kunsten, 2009, pp. 59–68.

[20] H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," *Prob. of Cont. and Inf. Th.*, vol. 15, no. 2, pp. 159–166, 1986.

[21] Y. X. Li, R. H. Deng, and X. M. Wang, "On the equivalence of mceliece's and niederreiter's public-key cryptosystems," *Information Theory, IEEE Transactions on*, vol. 40, no. 1, pp. 271 –273, jan 1994.

[22] E. Prange, "The use of information sets in decoding cyclic codes," *IRE Transactions on Information Theory*, vol. 8, no. 5, pp. 5–9, 1962.

[23] J. Stern, "A method for finding codewords of small weight," in *Coding Theory and Applications*, ser. LNCS, G. Cohen and J. Wolfmann, Eds., vol. 388. Springer, 1989, pp. 106–113.

[24] A. Becker, A. Joux, A. May, and A. Meurer, "Decoding random binary linear codes in $2^{n/20}$: How 1+1=0 improves information set decoding," in *Advances in Cryptology - EUROCRYPT 2012*, ser. LNCS, vol. 7237. Springer, 2012, pp. 520–536.

[25] N. Sendrier, "Decoding one out of many," in *Post-Quantum Cryptography*, ser. LNCS, B.-Y. Yang, Ed. Springer Berlin / Heidelberg, 2011, vol. 7071, pp. 51–67, 10.1007/978-3-642-25405-5-4.

[26] P.-A. Fouque and G. Leurent, "Cryptanalysis of a hash function based on quasi-cyclic codes," in *CT-RSA 2008*, ser. LNCS, T. Malkin, Ed., vol. 4964. Springer, 2008, pp. 19–35.

[27] P. Loidreau, personal communication.

[28] D. J. Bernstein, T. Lange, and C. Peters, "Attacking and defending the McEliece cryptosystem," in *2nd Int. Workshop on Post-Quantum Cryptography*, ser. PQCrypto'08. Berlin: Springer, 2008, pp. 31–46.